Korisniče površine: Operator interface, user interface

|Hardverske površine: grafička: ekran i alfanumerička:tastatura

Softverske površine: API (aplication process interface)

Vrste OS-a:

Monolitni sistemi (bilo koji program može pozvati bilo kojeg drugog (“opšti haos”) Imaju minimalni red

Modularni sistemi: grupisanje programa na osnovu toga koji koga poziva češće (u modul). Između modula postoji jako malo poziva

Samo programi u modulu znaju šta je u tom modulu.

Van modula se unutrašnji program poziva kroz interfejs

Tipični moduli: Za upravljanje procesima

Za upravljanje memorijom

Za upravljanje fajlove

UI modul (UI-ulazno izlazni)

bezbednosno zaštitni modul

Slojeviti sistemi: organizacija funkcija po slojevima

Funkcije mogu zvati funkcije samo iz sloja neposredno ispod njihovog, isto tako njih mogu samo zvati funkcije u sloju neposredno iznad njihovog

ako su slojevi daleko treba mnogo veza (vremena)

svaki sloj predstavlja virtuelnu mašinu koja je složenija od one ispod nje.

THE (Dijsktra)

Sloj funkcja: 5 upotrebni programi

4 korisničnki programi

3 UI (ulazno izlazni)

2 komunikacija procesa

1 memorija

0 raspoređujući

Virtuelna mašina: Osnovna ideja: jezgro (kernel) predstavlja jednu virt. mašinu.

procesi koriste i HW i jezgro.

vođenje evidencije o procesima, stanju HW-a i stanju kernela (jezgra)

CMS (conversational monitor system)

spoljašnje jezgro (egzojezgro)

Model klijent-server: delovi os-a rade kao korisnički programi

klijentski i serverski procesi

povećana pouzdanost

distribuiran sistem (raspodeljen na više uređaja)

Nijedan os nije čist oblik od navedenih vrsta nego mešavina.

Osnovni pojam svakog os je proces

pod procesom možemo smatrati program koji se izvršava

proces ima karakteristike o kojima os vodi računa

os može vršiti operacija sa ili nad procesima

program je uvek isti, a proces je dinamički objekt (menja se tokom izvršavanja)

Niti (threads): procesi se izvršavaju sekvencijalno

isti program se može izvršiti na različite načine.

Svakom procesu možemo pridružiti logički procesor i logičku memoriju

da bi os radio HW mora ispuniti neke preduslove:

Mehanizam prekida

zaštitni mehanizam adresiranja memorije (sprečava pogrešno adresiranje)

skup privilegovanih instrukija (nisu dostupni korisnicima)

maskiranje prekida (proces ne zna da je bio prekinut)

dodeljivanje procesora drugom procesu

pristup zaštićenim memorijskim registrima

izvršavanje UI operacija

zaustavljanje procesora

časovnik realnog vremena (za evidenciju o korišćenju resursa)

Delovi os-a:

prvi nivo obrade prekida

raspoređivač sistema (dispečer)

dodeljuje procesor procesima

uvek ima više zahteva nego resursa

rutine za interprocesne komunikacije

Slanje i primanje poruka

semaforske tehnike

imenovane cevi

rutine za korišćenje zajedničkih memorija

Kako os “vidi” procese?

procesor je nedeljiv resurs (jedan po jedan proces)

Svaki procesor ima tri fundamentalna memorijska dela:

izvršni kod

konstante

promenljive(veličina nije unapred poznata)

PCB (process control block)

preventivni os (os sa pretpražnjenjem)

Promena konteksta (okruženja u kojem se nalazi proces):

1. HW postavi brojač naredbi u stek...
2. HW prepiše novi brojač naredbi iz vektora prekida
3. os pozove asembler rutinu da prepiše registre
4. asembler rutina postavi novi stek
5. rutine prekida generišu poruku i šalju je
6. raspoređivač prebaci proces koji je na redu u stanje spreman za izvršivanje
7. isto kao 6 samo obrnuto
8. os promeni kod, vraća se asembler rutina
9. pokrene se novi kod

izvršavanje uputstava je uglavnom nedeljivo (ne sme se prekinuti izvršavanje jednog uputstva (usred uputstva))

Procesor ima svoje promenljive:

brojač

pokazivač

registre

signalne bitove

(njihove vrednosti utiču na izvršavanje sledećeg uputstva)

RAM model:

adrese su brojevi